IN THE CLAIMS

Please amend the claims as follows:

1.   (Currently Amended) A system comprising:

a database;

a message server having no persistent state such that the message server can be restarted after a failure without performing state recovery operations; and

a plurality of instances of an application server implementing a Java application model coupled in a star topology with the message server at a center of the star topology, the plurality of instances sharing the database, one or more of the plurality of instances to initiate registering or reregistering of instance-specific information with the message server upon starting or restarting, respectively, of the message server, the instance-specific information including a confirmation of a connection between the one or more of the plurality of instances and the message server.

2.   (Original) The system of claim 1 wherein each instance comprises:

a dispatcher node; and

a plurality of server nodes.

3.   (Original) The system of claim 2 wherein each server node comprises:

a java 2 enterprise edition (J2EE) engine.

4.   (Original) The system of claim 1 further comprising: ·

a central lock server to provide cluster wide locks to the plurality of instances.

5.   (Original) The system of claim 1 wherein the message server comprises:

a first data structure to store a list of connected clients; and

a second data structure and a list of services provided in the system.

6.   (Currently Amended) A computer readable storage media containing executable computer program instructions which when executed cause a digital processing system to perform a method comprising:

starting a central services node to provide a locking service and a messaging service, the messaging service having no persistent state;

starting a plurality of application server instances; ~~and~~

organizing the application server instances into a cluster having star topology with the central services node at a center of the star topology; and

registering or reregistering instance-specific information with the central services node upon starting or restarting, respectively, of the central services node, the registering or registering initiated by one or more of the plurality of application server instances, the instance-specific information including a confirmation of a connection between the one or more of the plurality of instances and the central services node.

7.   (Original) The computer readable storage media of claim 6 containing executable computer program instructions which when executed cause a digital processing system to perform the method further comprising:

sharing a database among the plurality of application server instances.

8.   (Original) The computer readable storage media of 6 containing executable computer program instructions which when executed cause a digital processing system to perform the method wherein starting a plurality of application server instances comprises:

starting, for each application server instance of the plurality, a dispatcher node and a plurality of server nodes.

9.   (Original) The computer readable storage media of claim 6 containing executable computer program instructions which when executed cause a digital processing system to perform the method further comprising:

starting a message server having no persistent state.

10.   (Canceled)

11.   (Original) The computer readable storage media of claim 6 containing executable computer program instructions which when executed cause a digital processing system to perform the method further comprising:

   conducting inter instance communication through the messaging service.

12.   (Original) The computer readable storage media of claim 9 containing executable computer program instructions which when executed cause a digital processing system to perform the method further comprising:

   restarting the message server without state recovery responsive to a system failure.

13.   (Original) The computer readable storage media of claim 10 containing executable computer program instructions which when executed cause a digital processing system to perform the method further comprising:

   notifying all registered instances from the message server when an additional instance joins the cluster.

14.   (Currently Amended) A system comprising:

   means for organizing a plurality of application servers instances into a cluster having a star topology with a central services node at a center of the star topology;

   means for sharing a storage resource across the cluster; and

   means for performing centralized inter instances communication without maintenance of persistent state information, the inter instances communication including registering or reregistering instance-specific information with the central services node upon a starting or restarting, respectively, of the central services node, the registering or reregistering initiated by one or more of the plurality of application server instances, the instance-specific information including a confirmation of a connection between the one or more of the plurality of application server instances and the central services node.

15.   (Original) The system of claim 14 further comprising:

means for centralized locking of a resource within the cluster.

16.   (Original) The system of claim 14 wherein the means for performing comprises:

a message server having no persistent state.

17.   (Currently Amended) The system of claim 14 wherein the means for performing

comprises:

~~means for registering instances; and~~ means for recording services provided in the cluster.

18.   (Currently Amended) A method comprising:

starting a central services node to provide a locking service and a messaging service, the

messaging service not maintaining a persistent state;

starting a plurality of application server instances; ~~and~~

organizing the application server instances into a cluster having star topology with the

central services node at a center of the star topology; and

registering or reregistering instance-specific information with the central services node

upon starting or restarting, respectively, of the central services node, the registering or

reregistering initiated by one or more of the plurality of application server instances, the

instance-specific information including a confirmation of a connection between one or more of

the plurality of application server instances and the central services node.

19.   (Original) The method of claim 18 further comprising:

sharing a database among the plurality of application server instances.

20.   (Original) The method of claim 18 wherein starting a plurality of application server

instances comprises:

starting, for each instance of the plurality, a dispatcher node and a plurality of server

nodes.

21.   (Original) The method of claim 18 wherein starting a central service node comprises:

starting a message server having no persistent state.

22.   (Canceled)

23.   (Original) The method of claim 18 further comprising:

conducting inter instance communication through the messaging service.

24.   (Original) The method of claim 21 further comprising:

restarting the message server without state recovery responsive to a system failure.

25.   (Original) The method of claim 22 wherein organizing further comprises:

notifying all registered instances from the message server when an additional instance

joins the cluster.

26.   (Canceled)

27.   (Previously Presented) The system of claim 1, wherein inter-instance communications

are conducted through the messaging server.

28.   (Previously Presented) The system of claim 26, wherein each registered application

server instance is notified by the message server when an additional instance registers with the

messaging server.

29.   (New) The system of claim 1, wherein each of the plurality of instances is started using a

first instance-specific bootstrap logic, the first instance-specific bootstrap logic synchronized

with a second instance-specific bootstrap logic stored in the database.

30.   (New) The system of claim 1, wherein a node within the plurality of instances is started using a first node-specific bootstrap logic, the first node-specific bootstrap logic synchronized with a second node-specific bootstrap logic stored in the database.

31.   (New) The method of claim 18, wherein the instance-specific information further includes information about a new service that the one or more of the plurality of instances provide.